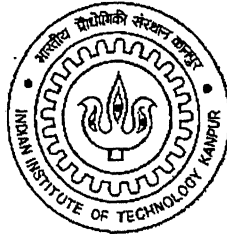


System for Diagnosing Valvular Heart Disease using Heart Sounds

*A thesis submitted
in partial fulfillment of the requirements
for the degree of
Master of Technology*

by

Soumya Sakha Tripathy



to the

**Department of Computer Science & Engineering
Indian Institute of Technology
Kanpur-208016, India**

June, 2005

Certificate

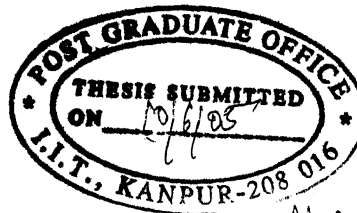
This is to certify that the work contained in the thesis entitled "*System for Diagnosing Valvular Heart Disease using Heart Sounds*", by *Soumya Sakha Tripathy*, has been carried out under my supervision and that this work has not been submitted elsewhere for a degree.

June, 2005

Harish

(Dr. Harish Karnick)

Department of Computer Science & Engineering,
Indian Institute of Technology,
Kanpur.



Shrotri

TH
CSE/2005/M
T 7378

19 JUL 2005/CSE
गुस्तात्म का गानाथ कलकर पुस्तकालय
भारतीय प्रौद्योगिकी संस्थान कानपुर
अपानि ३० १५१९८२



A151982

Abstract

Heart disease is a major cause for mortality today. *Heart auscultation* (the monitoring of sounds produced by heart), is a simple tool in the diagnosis of heart diseases, especially valvular diseases. It is particularly important in primary health care, due to its effectiveness in detecting a wide range of heart abnormalities, and to the low cost of the equipment involved. However, forming a diagnosis based on heart sounds is a skill that can take years to acquire. Particularly in remote areas and in developing country like ours, physicians with the necessary training may not be widely available.

This thesis presents a diagnosis system based on heart auscultation. A library of heart sound files, recorded via an electronic stethoscope are used, features from these samples are extracted using *discrete wavelet transform* and the classification is carried out by using a feed forward *neural network*. The performance of the system was satisfactory considering the paucity of data available for training.

**Dedicated to
My Parents and Teachers**

Acknowledgment

I take this opportunity to express my sincere gratitude, regards and thanks to all those who contributed in making my thesis a success.

First of all I would like to express my hearty gratitude towards my supervisor Dr. Harish Karnick for his excellent guidance, invaluable suggestions and generous help at all the stages of my research work. His interest and confidence in me was the reason for all the success and take my thesis to completion. He was the best adviser and teacher I could have wished for. In spite of his hectic schedule he was always approachable and took his time off to attend my problems and give the appropriate advice. I also wish to thank whole heartily all the faculty members of the Department of Computer Science and Engineering for the invaluable knowledge they have imparted to me and for teaching the principles in most exciting and enjoyable way.

I would like to give special thanks to Gita Pathak of *Media Lab Asia* for providing me the heart samples, which was the indispensable part of my thesis. She also gave many suggestions from time to time for carrying out my work.

I would also like to thank my classmates of the great *MTech2003* batch, and specially Rajiv, Madhav, Nishit, Ganga, Rohan, Nitya, Adi of MLA group for helping me and proving suggestions during every stage of my thesis.

My two years stay here was very pleasant and unforgettable. This is due to the warm presence of my classmates and my wingmates at hostel. I would like to thank all the *E-Toppers* of hall-7. Hochi-da, Kesta, Utsa, Anik, Nitin, Rahul, Kuldip deserve a special mention.

Last, but not the least, I would like to thank my alma-mater Kalyani Govt. Engg. College for deputing me to pursue my masters degree here, and my parents for taking me to this stage in my life.

Soumya Sakha Tripathy

Contents

1	Introduction	2
1.1	Heart Sounds	3
1.1.1	Sources of Heart Sounds	3
1.1.2	Heart Valve Problems	4
1.1.3	Heart Murmurs	5
1.1.4	Auscultation	5
1.2	Phonocardiogram	5
1.3	Problem Definition and Approach	6
1.4	Organization of the Report	7
2	Related Work	8
2.1	Previous Work	8
2.2	Our Contribution	11
3	Structure of Classification System	12
3.1	Data Acquisition and Pre-processing	12
3.1.1	Data Acquisition	12
3.1.2	Data Cleaning	13
3.1.3	Normalization	13
3.2	Feature Extraction	13
3.2.1	Fourier Transform	13
3.2.2	Short Time Fourier Transform	14
3.2.3	Wavelet Transform	16
3.3	Feature Selection	19
3.4	Classification	20
3.4.1	A Brief Introduction to Neural Networks	21
4	Implementation and Results	25
4.1	Display Subsystem	25
4.2	Diagnostic Subsystem	26
4.2.1	Data Denoising	27

4.2.2	Cycle Finding	27
4.2.3	Feature Extraction	30
4.2.4	Feature Selection	30
4.2.5	Classification	31
4.3	Results	33
5	Conclusion and Future Work	35
5.1	Conclusion	35
5.1.1	Problems faced	35
5.1.2	Advantages of our system	36
5.1.3	Drawbacks of the system	36
5.2	Future Work	36
	Bibliography	37

List of Figures

1.1	Two phases of a typical heart cycle : <i>systole</i> and <i>diastole</i>	4
1.2	Auscultation Areas	6
1.3	Phonocardiogram	6
3.1	Block diagram of the <i>training</i> module	12
3.2	Block diagram of the <i>testing</i> module	12
3.3	Short Window Fourier Transform	15
3.4	Comparative time and frequency resolution of STFT and wavelet transform	16
3.5	Daubechies mother wavelet	20
3.6	A typical <i>neural network</i> structure	21
3.7	A <i>sigmoid</i> unit	22
3.8	A multilayer feed-forward network	23
4.1	Block diagram of our classification system	25
4.2	The phonocardiogram generated by our system	26
4.3	Moving average algorithm	28
4.4	The <i>alpha</i> and <i>beta</i> regions in heart sound	29
4.5	Wavelet Decomposition by <i>pyramidal algorithm</i>	31

List of Tables

4.1	Daubechies' Wavelet Coefficients	32
4.2	Different Parameters	33
4.3	Classification Accuracy	33

Chapter 1

Introduction

According to the World Health Organization (WHO) heart disease and stroke kill some 17 million people a year, which is almost one-third of all deaths globally. By 2020, heart disease and stroke will become the leading cause of both death and disability worldwide.

“No matter what advances there are in high-technology medicine, the fundamental message is that any major reduction in deaths and disability from heart disease and stroke will come primarily from prevention, not just cure. We believe that early cardiac physical examination (screening) will become a fundamental element of any prevention program.” – Dr Judith Mackay, co-author of the WHO Atlas of heart disease and stroke.

So, it is very clear that proper diagnosis of heart disease is important for patients to survive. Physicians have to know the condition of the heart to decide for surgery or non-invasive treatment. Though *electrocardiogram* (ECG) is an important tool for diagnosis, it has some drawbacks like :

- it can detect diseases that are more or less related to blood-circulation and blood vessels, but there are heart diseases (structural abnormalities in heart valves and defects characterized by heart murmurs) that are difficult to detect using ECGs.
- cost of ECG equipment is high.
- limited availability of ECG equipment.
- special skill required to administer and interpret the results of ECG.

The problem is similar with the recently developed *echocardiography*, as it is bulky and expensive. Thus, in remote areas or in developing countries, *auscultation* (diagnosis through heart sounds) seems a feasible alternative. And it would be better if we can use both *echocardiograph* or ECG and *auscultation* to achieve even better diagnosis.

It is worthwhile to mention that, historically, the bare ear and the stethoscope were of great help in diagnosing most heart diseases, but it has been somewhat eclipsed in the research literature due to the advent of electrocardiographic methods. However, forming a diagnosis based on sounds heard through either a conventional acoustic stethoscope or an electronic one is itself a very special skill, and it may take years to acquire. Despite its obvious utility, because this skill is also very difficult to teach and grasp in a structured way, the majority of medicine and cardiology programs offer no such instruction. It would be very useful if the benefits of auscultation could be obtained with a simpler method, and using low-cost, easy to use equipment. In this thesis we report the design and testing of a digital auscultation system which can be used for heart sound based diagnosis. In addition, the system can be used for training medical students since heart sounds can both be heard and seen on the screen.

1.1 Heart Sounds

Heart sounds are complex and highly nonstationary signals. The “beats” associated with these sounds are reflected in the signal by periods of relatively high activity, alternating with comparatively long intervals of low activity. The heart beat usually has two sound components: Lub and Dub. Lub is the first sound and Dub is the second sound. They are also referred to as S1 and S2. These sounds follow each other in a cyclic fashion.

In addition to S1 and S2, which are always present, third and fourth heart sounds (S3 and S4) may also be heard. If present, S3 occurs shortly after S2. When S4 is audible, it occurs shortly before S1.

1.1.1 Sources of Heart Sounds

While the pathological origins of all contributions to these sounds (S1, S2 etc) are not agreed upon, it is clear that the closure of the heart valves are the major contributor.

Heart Valves

Blood is pumped through the heart in only one direction. Heart valves play key roles in this one-way blood flow, opening and closing with each heartbeat. Pressure changes behind and in front of the valves allow them to open their flap-like “doors” at just the right time, then close them tightly to prevent a backflow of blood. There are 4 valves in the heart:

- Tricuspid valve
- Pulmonary valve
- Mitral valve

- Aortic valve

How Normal Heart Valves Work

The heart is divided into four chambers. The upper chambers are called atria and the lower chambers are called ventricles. Each heart cycle has two phases. In the 1st phase called *systole*, blood without oxygen returns from the body and flows into the heart's upper-right chamber (the right atrium). From there, it is forced through the tricuspid valve into the lower-right chamber (the right ventricle). The right ventricle pumps the blood through the pulmonary valve and into the lungs. While in the lungs, the blood picks up oxygen. As the right ventricle is preparing to push blood through the pulmonary valve, the tricuspid valve closes to stop blood from flowing back into the right atrium.

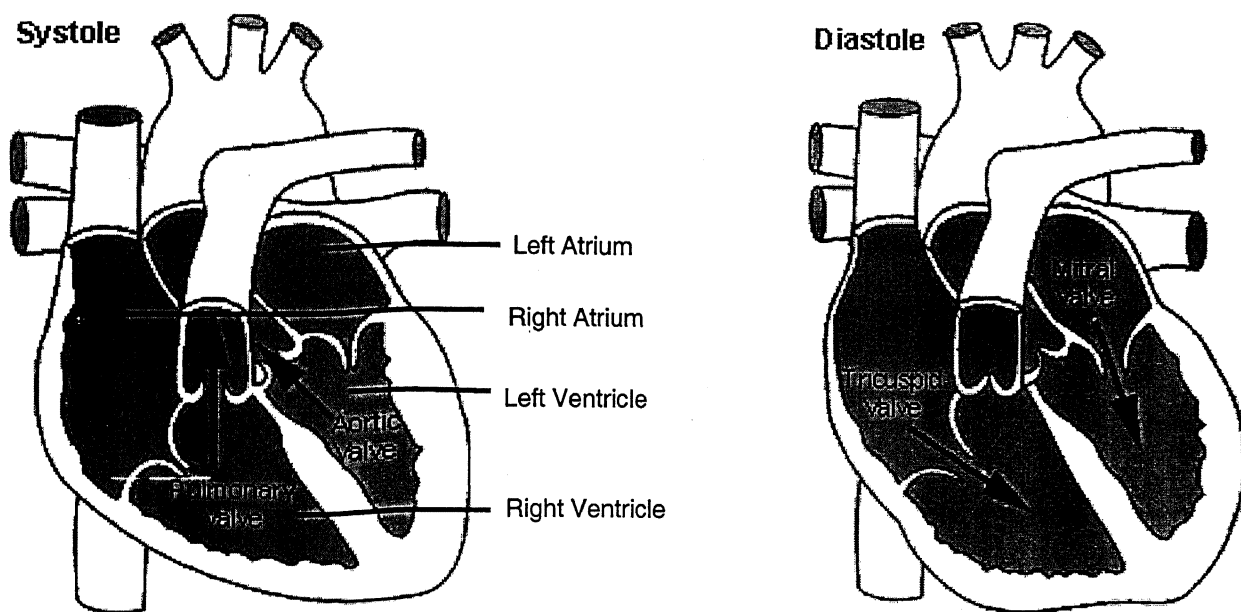


Figure 1.1: Two phases of a typical heart cycle : *systole* and *diastole*

In the next phase, *diastole*, oxygen-rich blood returning from the lungs flows into the upper-left chamber (the left atrium). This blood is forced through the mitral valve into the lower-left chamber (the left ventricle) with the mitral valve sealing off to stop the backflow of blood. At the same time that the right ventricle is pumping the blood without oxygen into the lungs, the left ventricle is pushing the blood with oxygen through the aortic valve and on to all of the body's organs.

1.1.2 Heart Valve Problems

Valve disease occurs when a valve does not work the way it should. If a valve does not open all the way, less blood can move through the smaller opening. If a valve does not close tightly,

blood may leak backward. These problems may mean the heart has to work harder to pump the same amount of blood. Or blood may back up in the lungs or body because it is not moving efficiently through the heart.

- **closing problem** : *insufficiency* (also called *regurgitation*) results when the valve doesn't close tightly. The valve's supportive structures may be loose or torn. Or the valve itself may have stretched or thinned. Blood then may leak back in the wrong direction through the valve.
- **opening problem** : *stenosis* occurs when a valve does not open completely. The valve may have become hardened or stiff with calcium deposits or scarring, so it is hard to push open. Blood has to flow through a smaller opening, so less blood gets through the valve into the next chamber or into the body.

In our thesis, we will mainly deal with these two kind of problems, namely *regurgitation* (or *insufficiency*) and *stenosis* of the 4 heart valves.

1.1.3 Heart Murmurs

A *heart murmur* is a swishing or a whistling sound that the doctor hears when he listens to a patient's heart. The doctor uses a tool called a stethoscope to listen to the heart.

A murmur is usually present when there is a heart valve problem. The doctor will perform a variety of tests to determine what kind of valve problem one has and whether the valve problem is serious. Some of the tests performed are: an echocardiogram, an electrocardiogram, a chest x-ray.

1.1.4 Auscultation

Auscultation is that part of the physical examination involving the act of listening with a stethoscope to sounds made by the heart and interpretation of the sounds. It is a technique that doctors have used since long. Generally, the stethoscope is moved over some specific areas in the chest to have the proper auscultation. The following figure¹ shows different auscultation areas.

1.2 Phonocardiogram

Phonocardiogram is the graphical representation of the sounds produced by the heart. It helps to visualize the different sound components present in a heart cycle and the time of their occurrence. Here is a snapshot of a phonocardiogram displayed by our system.

¹courtesy : www.3m.com

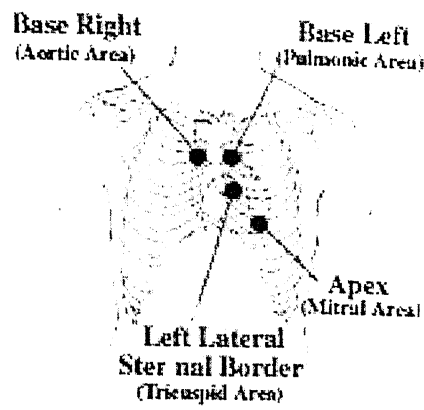


Figure 1.2: Auscultation Areas

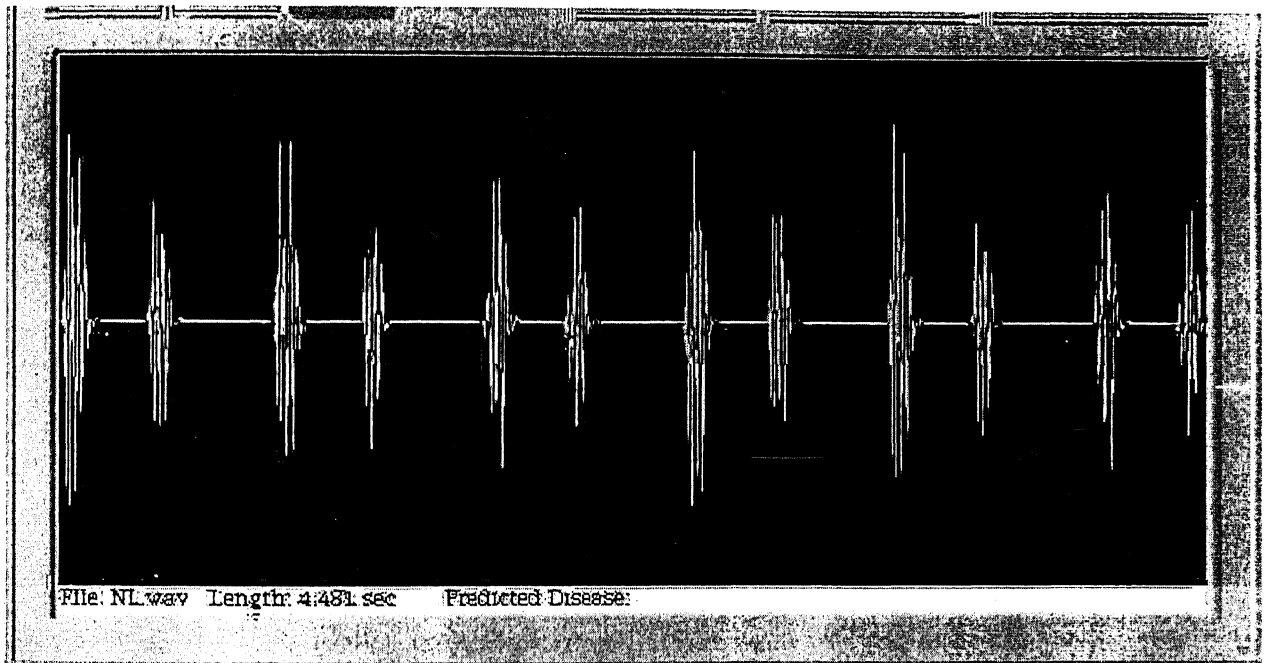


Figure 1.3: Phonocardiogram

1.3 Problem Definition and Approach

This thesis describes the development of an auscultation system. First we have to represent the heart sound (or auscultation) graphically, better known as *phonocardiogram*. This helps the doctor to hear the sound and visualize it simultaneously. The second part is concerned with a diagnosis system for valvular irregularities. This requires feature extraction and selection from the

phonocardiogram and then a classification algorithm to classify the irregular phonocardiograms. The irregularities we deal with are :

- Aortic Regurgitation
- Aortic Stenosis
- Mitral Regurgitation
- Mitral Stenosis

These diseases are all related to heart valves.

In our work, we use local signal analysis methods (*wavelet transform*) and classification techniques (*neural network*) to characterize and interpret sounds corresponding to symptoms important for diagnosis. It is hoped that the results of this analysis may prove valuable in itself as a diagnostic aid, and as input to more sophisticated machine diagnosis systems.

1.4 Organization of the Report

The report is organized as follows:

- In Chapter 2 we review existing work in this field, and then indicate our contribution in this area.
- In Chapter 3 we give an introduction to an intelligent heart sound classification system and discuss the different components of this system. These include feature extraction tool and the classification tool.
- In Chapter 4 we describe our system in detail. The different aspects of the implementation is covered here. We also discuss the results obtained using our system.
- Finally, Chapter 5 concludes and mentions possible extensions to this work.

Chapter 2

Related Work

When studying the research literature, we found some research on the use of phonocardiogram to diagnosis heart diseases. We also studied the work done on *Wavelet Transform* and *Neural Network* for feature extraction and classification of signals, because these are most relevant to our work. In the following section, the relevant literature is reviewed.

2.1 Previous Work

Ghosh, Deuser and Beck proposed a neural network based hybrid system for detection, characterization and classification of short-duration oceanic signals [6]. These signals are underwater signals obtained from passive sonar and contain valuable clues for source identification. After preprocessing (denoising) the signal they used multiresolution wavelets for finding the feature vector. Then three classifiers namely statistical classifier (kNN), ANN and recurrent network were used to classify those signals. Finally, the outputs of different classifiers were combined. In their experiment they obtained almost 100% accuracy.

Nguyen, Hammel and Gong used a system of multiple hybrid neural network [19] to classify contact signals recorded in open ocean sites. They used self-organizing feature map and neural network for classification. They used 1800 samples for training and 900 samples for testing. In this, they obtained almost 100% accuracy for classification.

Hippenstiel and Fargues used wavelets [2] to extract features from digital communication signals. We have studied the above works because, the signal they are dealing with, is similar in nature to phonocardiogram signals, in terms of transient behaviour.

In the late 80's Abdalla S. A. Mohamed and Hazem M. Raafat at the University of Regina, Canada did some work on recognition of heart sounds and murmurs for cardiac diagnosis [12, 13]. They developed a mathematical model to describe the heart sounds and murmurs by a finite number of parameters. An autoregressive model was selected to represent the heart sound at principal locations of cardiac auscultation and for different heart diseases. Feature extraction of the signals, based on fourth order linear prediction of the cardiac cycle frames was performed.

Then based on the minimum distance between the features of the measured pattern and reference patterns, classification was carried out.

With the advent of electrocardiogram (ECG) the focus shifted to ECGs. People tried to classify ECGs and predict the heart diseases accordingly. J. Zhu, N. Hazarika, A. C. Tsoi and A. A. Sergejew have worked [14, 15, 1] on this topic. They selected three types of ECG signals: Normal, Schizophrenia (SCH) and Obsessive Compulsive Disorder (OCD). Wavelet transform was used for feature extraction and a three-layered feedforward network which implements the backpropagation algorithm was used for learning. The system was able to classify over 66% of the normal class and 71% of schizophrenia class of ECG's.

Though the research on diagnosis of heart disease using heart sounds, was somewhat eclipsed by ECGs, work on heart sounds still continued.

In 1997, Huiying, Sakari and Iiro developed a heart sound segmentation algorithm [8]. The algorithm separates the heart sound signal in four parts: the first heart sound, the systolic period, the second heart sound and the diastolic period. They used digital heart sounds recorded on a multimedia PC equipped with an electronic stethoscope. First phonocardiograms of these heart sounds are created. Then using multi-level wavelet decomposition and reconstruction, the detail and approximation components of the phonocardiograms were extracted. They marked the locations where the signal value exceeded some selected threshold. Thus, they identified the S1s and S2s and based on that they segmented the signals. The performance of the algorithm was evaluated using 1165 cardiac periods from 77 digital phonocardiographic recording including normal and abnormal heart sounds. It showed over 93 percent accuracy.

Later in 2001, Lee, Kim and Hong proposed some methods for heart sound recognition [9]. They used three recognition techniques and compared the result. The first method recognizes the characteristics of heart sound by integrating important peaks and analyzing statistical variables in the time domain. The second method builds a database by principal component analysis on a set of training heart sounds in time domain. Later, this database is used for recognizing new heart sounds. The third method builds a similar database, but in time-frequency domain. They tried to classify the heart sounds into seven classes. It was noticed that the third method outperformed the others.

Finally, some works closely related to our thesis are as follows. M. S. Obaidat and M. M. Matalgah studied the performance of the short-time Fourier transform and wavelet transform to phonocardiogram(PCG) signal analysis [18]. They compared the performance of FT, STFT and wavelet transform. They found that wavelet transform is capable of detecting the two components, aortic valve component A2 and pulmonary valve component P2, of the second heart sound (S2) of a normal PCG signal which can not be detected by FT and STFT.

In 2001, Todd R. Reed, Nancy E. Reed and Peter Fritzson worked on analysis of heart sounds for symptom detection [17]. First, heart sounds were segmented (manually) into sample segments, each consisting of a single heartbeat cycle. Then each segment was transformed using wavelet decomposition, based on Coifman 4th order wavelet. The transformed vector was reduced to smaller vector size, by discarding levels with shortest scale. Finally, each vector was classified using a three layer neural network. The system was evaluated using heart sounds corresponding to five different conditions. It gave 100% accuracy for all heart sounds.

In the year 2002, Ibrahim Turkoglu, Ahmet Arslan and Ilkay Erdogan presented an intelligent pattern recognition system to diagnose mitral valve diseases [3] using Doppler signals. The Doppler signals of the mitral valve were obtained by placing a transducer over the chest of the patient with the aid of ultrasonic image. Wavelet packet decomposition was used to extract the features, then classification was carried out using a neural network. The performance of the system was evaluated on 105 samples that contained 39 normal and 66 abnormal subjects. The accuracy obtained was almost 94% for normal and abnormal subjects.

Onsy Abdel-Alim, Naddar Hamdy and Mohammed A. E. developed a system for heart disease diagnosis using heart sounds [16]. A 1-minute long record of heartbeats (obtained from faculty of medicine, Ain Shams University, Egypt) is used to extract features. They chose two sets of features. The first set consists of features like duration of first and second heart sounds, their ratio and difference etc. The second set of features was obtained by using Daubechies wavelet transform. The location of the stethoscope on the chest during recording was also considered as an additional feature. A feed forward neural network was used for classification. They used 650 cases for training the network and 200 cases for testing. A recognition rate of 95% was obtained in this case.

In 2003, M. L. Jacobson did his work on analysis and classification of physiological signals [7]. In particular he worked on *heart rate variability* (HRV) signals, which have been shown to contain diagnostic information on the condition of a patient's cardiac and circulatory system. He used wavelet transform decomposition (12th order Daubechies wavelet) as a means of signal characterization. Classification was implemented through cluster assignment based on *Euclidean distance*. The class centers were computed as the average of the known patient conditions. The unknown patient condition was classified to the class whose center was closest. He used only two diseases namely coronary heart disease (CHD) and diabetes mellitus (DM) in his work and the accuracy obtained was 100% and 97% for DM and CHD respectively.

2.2 Our Contribution

We have developed a system for display and diagnosis of heart auscultation. It helps to visualize the heart signals graphically. There is also provision for listening to the sound, where a cursor moves over the display in synchrony with the audio. This helps the users (mainly doctors) in better identifying the different sound components. Our system also helps to diagnose heart diseases, by analysing and classifying the heart sound signals.

In the previous section, it was found that some work was done on analysis and classification of heart sound signals. But, we noticed that most of the work skipped the job of finding a heart-beat cycle from a heart sound signal of multiple cycles. Some of them started their work on already extracted cycles, and the others extracted the cycles manually. We have developed a technique that can extract a cycle from a longer heart sound signal. We also noticed that due to improper recording environment (not studio-like), lots of noise was added to the sound samples. So, we used some data-smoothing techniques to filter out noise (to some extent).

It is well-established that the wavelet transform is most suitable for the analysis of transient signals like heart sounds. So, we use *Daubechies* wavelets to extract features. We compare the results given by D4, D6 and D8 wavelets and choose the best one. Then using a back-propagation neural network we classify the heart sound signals into different categories.

Chapter 3

Structure of Classification System

Classification is the process of assigning a label to an unknown pattern so that it is categorized into one of several known categories. In our work, we wish to classify heart sound samples into (probable) heart-disease categories. In this chapter, the theoretical foundations for the pattern recognition and classification system used in our study are discussed.

The following figures represents the block diagrams of a typical classification system, which consists of training and testing module.

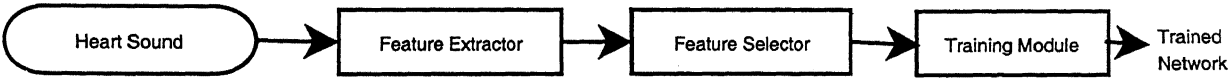


Figure 3.1: Block diagram of the *training* module

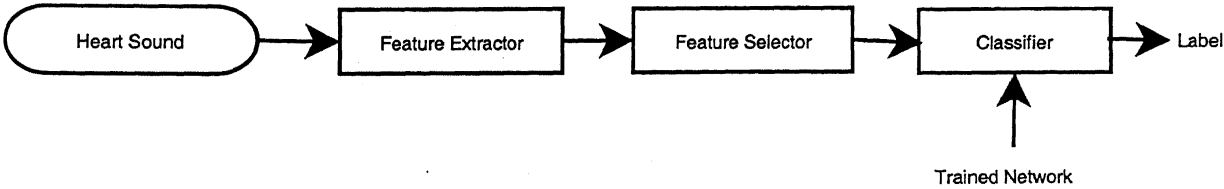


Figure 3.2: Block diagram of the *testing* module

3.1 Data Acquisition and Pre-processing

3.1.1 Data Acquisition

The first step is data acquisition. Data (or samples), that are to be used by the system, are collected. An electronic stethoscope is used to record the heart beats of different patients and

these recordings are then used for training the classifier.

The raw data collected may not always be usable by the system. There can be noise in the data, some data values may be missing etc, so the data is preprocessed.

3.1.2 Data Cleaning

Although most classification algorithms have some mechanisms for handling noisy or missing data, this step can help reduce confusion during learning.

Generally, noise removal is done by using *filtering techniques* or through *data smoothing techniques*. Missing values are replaced with the most commonly occurring value for that attribute or with the most probable value based on statistics.

3.1.3 Normalization

Data should also be normalized, particularly when neural network or methods involving distance measurement are used in the learning step. Normalization involves scaling all values for a given attribute so that they fall within a small specified range, such as -1.0 to 1.0, or 0.0 to 1.0. In methods that use distance measurements, for example, this would prevent attributes with initially large ranges from outweighing attributes with initially smaller ranges (such as binary attributes).

3.2 Feature Extraction

This is arguably the most important component of designing the classification system, since even the best classifier will perform poorly if the features are not chosen well. A feature extractor should reduce the pattern vector (i.e. the original waveform) to a lower dimension, which contains most of the useful information from the original vector. Thus it converts patterns to features in a condensed representation. Ideally, it should give only "relevant or important" information.

We briefly discuss 3 feature extraction techniques that are widely used for time-varying signals. These are *Fourier Transform*, *Short-Time Fourier Transform* and *Wavelet Transform*.

3.2.1 Fourier Transform

Joseph Fourier showed that any 2π -periodic function $f(x)$ can be expressed as the sum of a possibly infinite series of sines and cosines. The sum is also referred to as a Fourier expansion.

$$f(x) = a_0 + \sum_{k=1}^{\infty} (a_k \cos kx + b_k \sin kx)$$

The coefficients a_0 , a_k and b_k are calculated by

$$a_0 = \frac{1}{2\pi} \int_0^{2\pi} f(x) dx$$

$$a_k = \frac{1}{\pi} \int_0^{2\pi} f(x) \cos(kx) dx$$

$$b_k = \frac{1}{\pi} \int_0^{2\pi} f(x) \sin(kx) dx$$

The Fourier Transform's ability lies in its ability to analyze a signal in the time domain for its frequency content. The transform works by first translating a function in the time domain into a function in the frequency domain. The signal can then be analyzed for its frequency content because the Fourier coefficients of the transformed function represents the contribution of each sine and cosine function at each frequency.

However, the big disadvantage of a Fourier Expansion is that it has only frequency resolution and no time resolution. This means although we might be able to determine all the frequencies present in the signal, we do not know when they are present. This is because sines and cosines which comprise the bases of Fourier transform, are non-local and stretch out to infinity. They are therefore very poor in approximating data with sharp discontinuities.

Before moving to the next section, let's briefly discuss about *stationary* and *non-stationary* signals. A stationary signal is one whose frequency content does not change over time. So, in case of stationary signals, one does not need to know at what times which frequency components exist, since all frequency components exist at all times. But, this is not the case with non-stationary signals. Here, the frequency content changes over time. Some frequencies which are present in a particular time instance, may not be present later. Thus, finding the frequency components and their time of occurrence in a non-stationary signal becomes a challenging job.

3.2.2 Short Time Fourier Transform

The problem with Fourier transform was that it did not work for non-stationary signals. Now, can we treat some portion of a non-stationary signal as stationary? If this region where the signal can be assumed to be stationary is too small, then we look at that signal through narrow window, narrow enough that the portion of the signal seen from the window is indeed stationary. This approach ended up with a revised version of the Fourier transform, called *Short Time Fourier Transform* (STFT).

There is only a minor difference between STFT and FT. In STFT, the signal is divided into small enough segments, where these segments of the signal can be assumed to be stationary.

For this purpose, a window function “w” is chosen. The width of this window must be equal to the segment of the signal where stationarity is valid. This window function is first located at the very beginning of the signal. That is, the window function is located, centered with $t=0$. Let us suppose that the width of the window is “T” second. At the time instant $t=0$, the window function will overlap with the first $T/2$ seconds. The window function and the signal are then multiplied. By doing this, only the first $T/2$ seconds of the signal is being chosen. Then this product is assumed to be just another signal, whose FT is to be taken. The result of this transformation is the FT of the first $T/2$ seconds of the signal. If this portion of the signal is stationary, as assumed, then there will be no problem and the obtained result will be a true frequency representation of the first $T/2$ seconds of the signal. The next step, would be shifting this window (for some t_1 seconds) to a new location, multiplying with the signal, and taking the FT of the product. This procedure is followed, until the end of the signal is reached shifting the window by “ t_1 ” seconds at each step. The following figure explains this process.

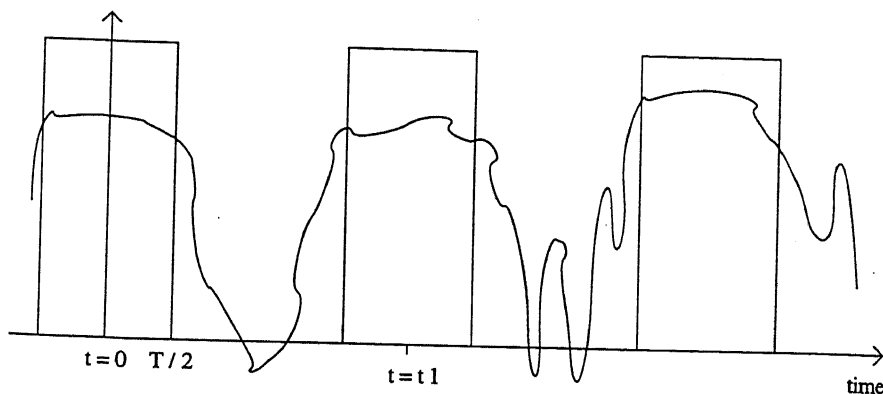


Figure 3.3: Short Window Fourier Transform

The following definition of the STFT summarizes the above explanation :

$$STFT(t', f) = \int_t [x(t)w^*(t - t')]e^{-j2\pi ft} dt$$

Here, $x(t)$ is the signal itself, $w(t)$ is the window function, and $*$ is the complex conjugate. One can see that the STFT of the signal is nothing but the FT of the signal multiplied by a window function. Here, for every t' (time) and f (frequency), a new STFT coefficient is computed.

So, now we have a true time-frequency representation of the signal. We not only know what frequency components are present in the signal, but we also know where they are located in time. But there are problems with STFT too. The problem has something to do with the width of the window function that is used. In the last section we have seen that FT has no resolution problem in the frequency domain, i.e. we know exactly what frequencies exist. And the time resolution in FT is zero, since we have no information about time. What gives the perfect frequency resolution in FT is the fact that the window used in the FT is its kernel, the sine and cosine function, which

lasts at all times from $-\infty$ to $+\infty$. Now, in STFT, our window is of finite length, thus it covers only a portion of the signal, which causes the frequency resolution to get poorer. That means we no longer know the exact frequency components that exist in the signal, but we only know a band of frequencies that exist.

Thus here comes a trade-off. If we use a window of infinite length, we get the FT, which gives perfect frequency resolution, but no time information. Furthermore, in order to obtain the stationarity, we have to have a short enough window, in which the signal is stationary. The narrower we make the window, the better the time resolution, and better the assumption of stationarity, but poorer the frequency resolution. The *wavelet transform* solves this trade-off problem.

3.2.3 Wavelet Transform

A wavelet allows one to do *multi-resolution analysis*, which helps to achieve both time and frequency localization. Here, the scale (or resolution, actually it is inverse of frequency) that we use to look at data plays a vital role. Wavelet algorithms process data at different scales or resolutions. If we look at a signal with a large "window", we would notice gross (or averaged) features. Similarly, if we look at a signal with a small "window", we would notice detailed features. Thus, by using varying resolution, it solves the problem that was there with STFT, due to the use of fixed window size (or resolution).

The following figure compares the relative frequency and time domain resolution of STFT and wavelet transform :

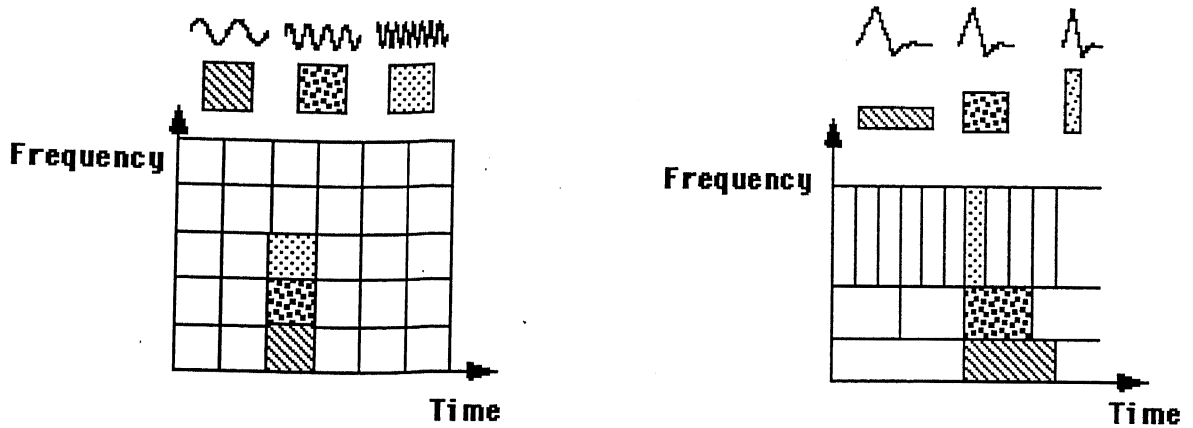


Figure 3.4: Comparative time and frequency resolution of STFT and wavelet transform

At the core of a wavelet analysis procedure is the choice of a wavelet prototype function, called a *mother wavelet*. Temporal analysis is performed with a contracted, high-frequency version of the prototype wavelet, while frequency analysis is performed with a dilated, low-frequency version of

the same wavelet. Because the original signal can be represented in terms of a wavelet expansion (using coefficients in a linear combination of the wavelet transforms), data operation can be performed using just the corresponding wavelet coefficients.

A Brief Introduction to Wavelets

The word "wavelet" literally means "small wave". We have seen that the basis functions of Fourier Transform are sine and cosine waves, which extend over the entire time axis from $-\infty$ to $+\infty$. That is the reason why FT can not provide time resolution. But, this is not the case with wavelets. Wavelets are localized waves and they extend not from $-\infty$ to $+\infty$ but only for a finite duration. So, they can provide both time and frequency resolution.

Wavelets were first introduced in the work of A. Haar (1909). One property of the Haar wavelet is that it has *compact support*, which means that it vanishes outside of a finite interval. But, Haar wavelets are not continuously differentiable. This somewhat limits the application of Haar wavelet.

In the 60's and 70's R. Coifman did some study on wavelets. Later in 1980, Grossman and Morlet defined wavelets in the context of quantum physics. In 1985, Stephen Mallat used wavelet for digital signal processing. He discovered some relationship between quadrature mirror filters, pyramid algorithms and orthonormal wave bases (we discuss later about these). Inspired by this work, Y. Meyer constructed the first non-trivial wavelets. Unlike Haar wavelet, the Meyer wavelets are continuously differentiable; however they do not have compact support. A couple of years later, Ingrid Daubechies used Mallat's work to construct a set of orthonormal basis functions that are most elegant. These functions are mostly used in today's wavelet applications. In our work, we use Daubechies' Wavelets.

Continuous Wavelet Transform

Continuous wavelet transform can be formally written as :

$$\gamma(s, \tau) = \int f(t) \psi_{s,\tau}^*(t) dt$$

The * denotes complex conjugation. This equation shows how a function $f(t)$ is decomposed into a set of basis functions $\psi_{s,\tau}(t)$, called wavelets. The variables s and τ , scale and translation, are the new dimensions after the wavelet transform.

The wavelets are generated from a single basic wavelet $\psi(t)$, the so-called *mother wavelet*, by scaling and translation.

$$\psi_{s,\tau}(t) = \frac{1}{\sqrt{s}} \psi\left(\frac{t-\tau}{s}\right)$$

Here, s is the scale factor, τ is the translation factor and the factor $s^{-1/2}$ is used for energy normalization across the different scales.

It should be noted that in the above equations the wavelet basis functions are not specified. This is the main difference between the Fourier transform and the wavelet transform. The theory of wavelet transform deals with the general properties of wavelet. Thus it defines a framework, based on which one can design the wavelet he wants.

Discrete Wavelet Transform

The continuous wavelet transform (CWT) described in the last section has redundancy. CWT is calculated by continuously shifting a continuously scalable function over a signal and calculating the correlation between them. It is clear that these scaled functions will be nowhere near an orthonormal basis and the obtained wavelet coefficients will therefore be highly redundant. To remove this redundancy Discrete Wavelet Transform (DWT) is used. In DWT the scale and translation parameters are chosen such that the resulting wavelet set forms an orthogonal set, i.e. the inner product of the individual wavelets $\psi_{s,\tau}$ are equal to zero.

Discrete wavelets are not continuously scalable and translatable but can only be scaled and translated in discrete steps. This is achieved by modifying the wavelet representation as

$$\psi_{s,\tau}(t) = \frac{1}{\sqrt{s_0^s}} \psi \left(\frac{t - \tau \tau_0 s_0^s}{s_0^s} \right)$$

Here s and τ are integers and $s_0 > 1$ is a fixed dilation step. τ_0 is the translation factor and it depends on the dilation step. The effect of discretizing the wavelet is that the time-scale space is now sampled at discrete intervals. We generally choose $s_0 = 2$ so that the sampling of the frequency axis corresponds to *dyadic sampling*. For the translation factor we generally choose $\tau_0 = 1$. In that case the previous equation becomes :

$$\psi_{s,\tau}(t) = \frac{1}{\sqrt{2^s}} \psi \left(\frac{t - \tau 2^s}{2^s} \right)$$

One of the most useful features of wavelets, specially for engineers, is that we can choose the defining coefficients for a given wavelet system to be adopted for a given problem.

From a signal processing point of view, wavelet transforms are like filter banks. Applying wavelet transform has the same effect of applying a filter i.e. we get detail or average (smoothed) component of the original signal. So, the wavelet coefficients corresponding to a particular wavelet, are called *wavelet filter coefficients*. The filter coefficients are placed in a transformation matrix, which is applied to a raw data vector. The coefficients are ordered using two dominant patterns, one that works as a smoothing filter(like a moving average) and one pattern that gives the detailed information. The following example will simplify the idea.

Consider a filter with four coefficients, c_1, \dots, c_4 . In this case, the transformation matrix becomes the following, which acts on a vector of data.

$$\begin{bmatrix} c_1 & c_2 & c_3 & c_4 & 0 & 0 & \dots \\ d_1 & d_2 & d_3 & d_4 & 0 & 0 & \dots \\ 0 & 0 & c_1 & c_2 & c_3 & c_4 & \dots \\ 0 & 0 & d_1 & d_2 & d_3 & d_4 & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}$$

The first row generates one component of data convolved with filter coefficients c_1, \dots, c_4 . Likewise the third, fifth and other odd rows. The even rows perform a different convolution, with coefficients d_1, \dots, d_4 (which are generated using c_1, \dots, c_4). The action of the matrix, overall, is thus to perform two related convolutions, then reduce each of them by half.

The filter c_1, \dots, c_4 acts as a smoothing filter, something like a moving average of four points. And the filter d_1, \dots, d_4 gives the details. The d 's are so chosen that, they give a zero response to a sufficiently smooth data vector. These two orderings of the coefficients are called a *quadrature mirror filter pair* from the signal processing point of view. In the next chapter we will explain the implementation of the above process in more detail.

Daubechies Wavelet

There are several types of wavelets like : Haar wavelet, Daubechies wavelet, Koifmann wavelet etc. The different wavelet families make different trade-offs between how compactly the basis functions are localized in space and how smooth they are. Daubechies wavelet [4, 5] seems most suitable for applications which deal with physiological signals like heart sounds etc. So, we use Daubechies wavelets in our work. The following figure ¹ shows one Daubechies mother wavelet. The inset figure shows its fractal nature.

Daub4, Daub6 and Daub8 :

Within each family of wavelets (e.g. Daubechies family) are wavelet subclasses distinguished by the number of coefficients and by the level of iteration. Often wavelets are classified within a family by the *number of vanishing moments*. This is a mathematical relationship that the coefficients must satisfy, and is directly related to the number of coefficients. There are many Daubechies wavelets like this. Among them most common are *Daub4*, *Daub6*, *Daub8* with 4, 6 and 8 coefficients respectively.

3.3 Feature Selection

Generally, the features selected in the previous step, are huge in number, often in the order of thousands. Also they have much redundancy in the sense that many of them do not carry much

¹<http://www.amara.com/IEEEwave/IEEEwavelet.html>

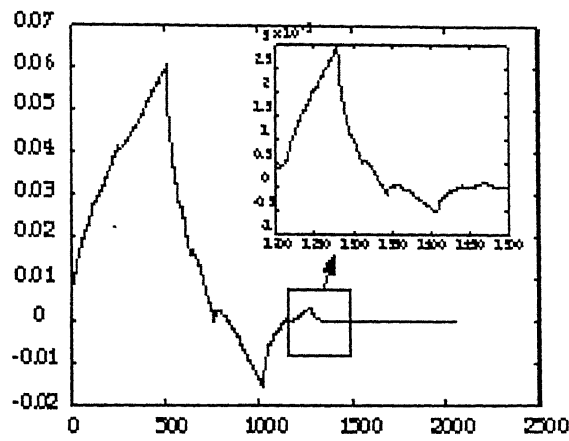


Figure 3.5: Daubechies mother wavelet

meaning and do not contribute in the decision of classification. So, they increase the computation cost without contributing to categorization. Hence, it is better to remove those features. Thus, the feature selection step deals with identifying a smaller number of meaningful features that best represent a given pattern without much redundancy.

There are many techniques that are used for selection. Some of these are:

- Discarding the values that are insignificant (small or zero). Generally a threshold level is chosen and all values smaller than the threshold are discarded.
- Choosing the top m large values from a set of n values.

3.4 Classification

Classification is the step where a specific pattern is assigned a specific class label according to the characteristic features selected for it. Many techniques are available for classification.

- Neural Network
- Decision Tree
- k - Nearest Neighbor (kNN)
- Case-Based Reasoning

We have used neural network in our system because of its suitability over other techniques. The problem with decision tree is its sensitivity to noise. This results in *overfitting* and hence wrong generalization. Case-based reasoning deals with complex symbolic descriptions of samples or “cases”, which is not applicable to our data set. The problem with kNN is that, unlike ANN,

it assigns equal weight to each attribute. This degrades its performance when there are many irrelevant attributes in the data set. Another problem with kNN is that, it stores all the training samples and do not build a classifier until a new sample needs to be classified. This results in longer testing time.

3.4.1 A Brief Introduction to Neural Networks

Artificial neural networks (ANN) provide a robust, general and practical method for learning real-valued, discrete-valued or vector-valued functions from examples (or samples). While ANNs are loosely motivated by biological neural systems, there are many complexities in biological neural systems that are not modeled by ANNs.

Roughly speaking, a neural network is a set of connected input/output units organized into layers, the geometry and functionality of which have been likened to that of the human brain. Each connection between the units has a weight associated with it. During the learning phase, the network learns by adjusting the weights so as to be able to predict the correct class label of the input samples. The following figure depicts a general neural network system:

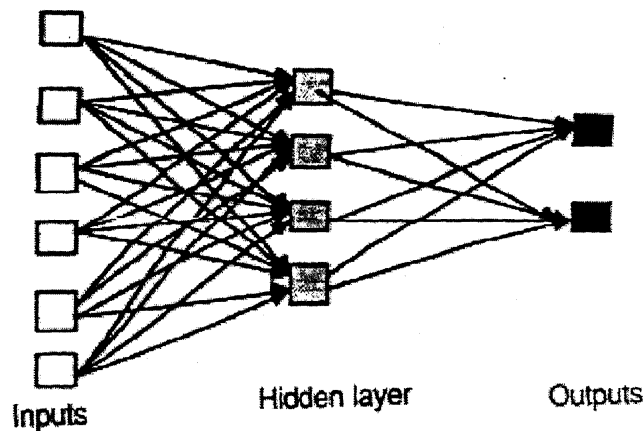


Figure 3.6: A typical *neural network* structure

Neural network involves long training time and therefore are more suitable for applications where this is feasible. They also require a number of parameters that are typically best determined empirically, such as network topology etc. The biggest advantage of neural network is the fact that they have high tolerance to noisy data.

The simplest Neural Networks are based on a *Perceptron unit*. But the problem with a perceptron is that a single perceptron can only express linear decision surfaces.

Multilayer Network and the Back-Propagation Algorithm

To learn non-linear decision surfaces, a new model is adopted. This is a multilayer neural network together with a *backpropagation* learning algorithm. Often a *sigmoid unit* is taken as the building block of this model. A sigmoid unit is very much like a perceptron, but based on a smoothed, differentiable activation function. It first computes a linear combination of its input, then applies the activation function to the result. The output o can be written as

$$o = \sigma(\vec{w} \cdot \vec{x})$$

where

$$\sigma(y) = \frac{1}{1 + e^{-y}}$$

σ is called the sigmoid function and its output ranges between 0 and 1, increasing monotonically with its input. The following figure shows a sigmoid unit.

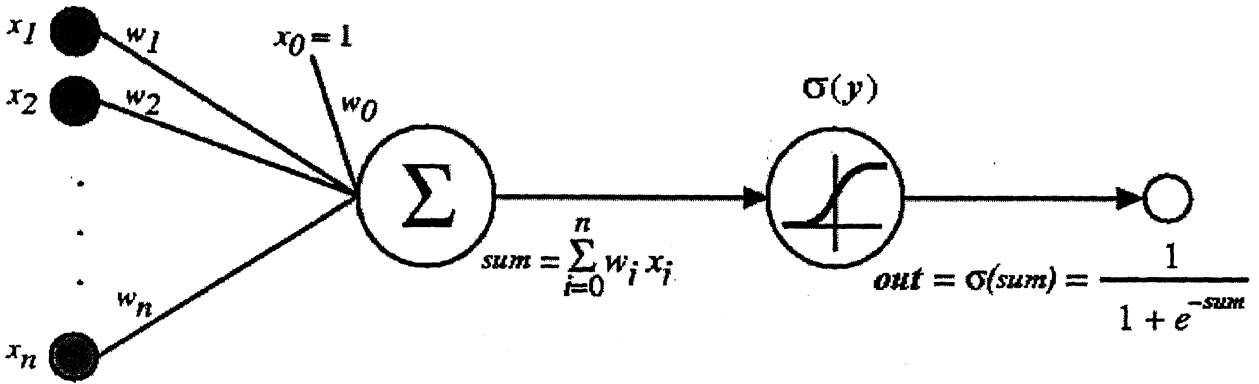


Figure 3.7: A *sigmoid unit*

Figure 3.8 shows a multilayer feed-forward network. The input corresponds to the attributes measured for the training example. The inputs are fed simultaneously into a layer of units making up the *input layer*. The weighted output of these units, are in turn, fed simultaneously to a second layer, called *hidden layer*. The hidden layer's weighted output can be input to another hidden layer, and so on. The number of hidden layers is arbitrary, although in practice, usually only one is used. The weighted output of the last hidden layer are fed to the *output layer*, which emits the network's prediction for the given samples. The network is called *feed-forward* because none of the weights cycles back to an input unit.

It should be noted, that, though the number of units in the input layer is determined by the number of input features and number of units in the output layer is determined by the number

of possible outcomes, there is no clear rule for determining the number of units in the hidden layer. So a trial-and-error method is adopted and the number that gives the best accuracy is chosen.

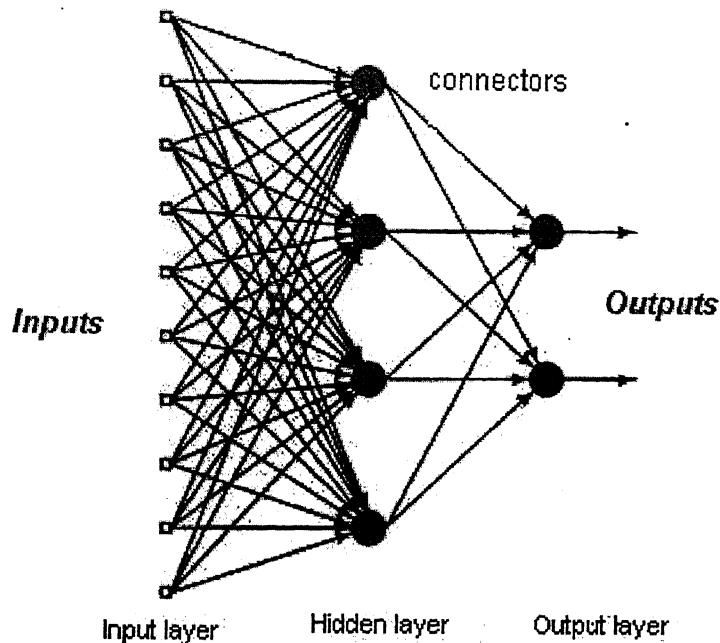


Figure 3.8: A multilayer feed-forward network

Learning Algorithm

Backpropagation learns by iteratively processing a set of training samples, comparing the network's prediction for each sample with the actual known class label. For each training sample, the weights are modified so as to minimize the mean square error between the network's prediction and the actual class. The modifications are made in the backward direction, i.e. from the output layer, through the hidden layer, to the input layer. That's why they are called "backpropagation".

The following is the backpropagation algorithm for the feed forward network containing two layers of sigmoid units [10]. It uses *gradient descent* as the training rule.

Here, each training example is a pair of the form (\vec{x}, \vec{t}) , where \vec{x} is the vector of network input values, and \vec{t} is the vector of target network output values. η is the learning rate (a small positive number, say 0.01). n_{in} is the number of network inputs, n_{hidden} is the number of units present in the hidden layer and n_{out} is the number of output units. The input from unit i into unit j is denoted as x_{ji} , and the weight from unit i to unit j is denoted as w_{ji} .

-
- Create a feed-forward network with n_{in} inputs, n_{hidden} hidden units and n_{out} output units.
 - Initialize all network weights to small random numbers.
 - Until the terminating condition is met, Do

a) For each (\vec{x}, \vec{t}) in *training_example*, Do

* *Propagate the inputs forward through the network:*

1. Input the instance \vec{x} to the network and compute the output o_j of every unit j in hidden or output layer of the network.

Propagate the errors backward through the network:

2. For each network output unit k , calculate its error δ_k

~~$$\delta_k = o_k(1 - o_k)(t_k - o_k)$$~~

3. For each hidden unit h , calculate its error term δ_h

$$\delta_h = o_h(1 - o_h) \sum_{k \in \text{outputs}} w_{kh} \delta_k$$

4. Update each network weight w_{ji}

$$w_{ji} = w_{ji} + \Delta w_{ji}$$

where

$$\Delta w_{ji} = \eta \delta_j x_{ji}$$

The commonly used terminating conditions are :

- when the classification error or the percentage of samples misclassified in the previous iteration is below some threshold.
- when a fixed number of iterations have happened.

Chapter 4

Implementation and Results

In the previous chapter, we have talked about various building blocks (or stages) of a classification system. Here, we will briefly describe how those stages have been implemented for the heart sound classification problem. The following figure describes the block diagram of our classification system:

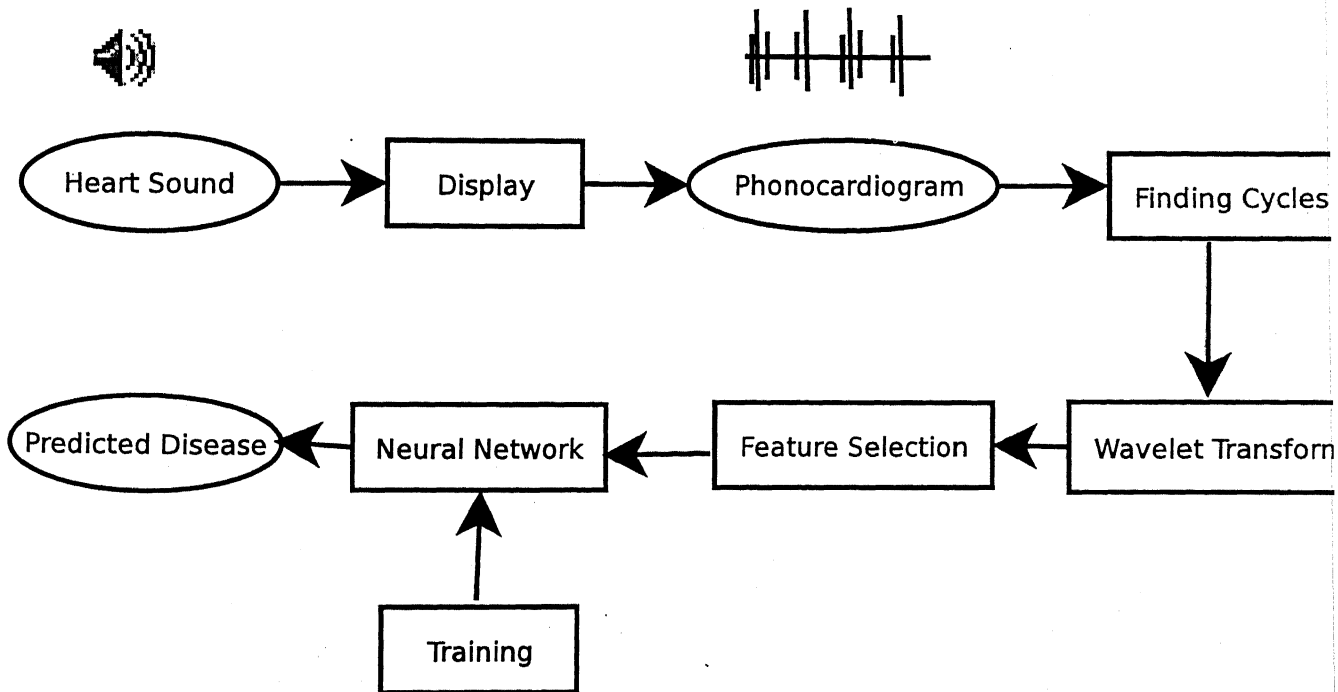


Figure 4.1: Block diagram of our classification system

4.1 Display Subsystem

We have implemented the whole system in *Java* to make it platform independent. The system displays the phonocardiogram of the heart sound. It also has other options like *Play*, *Show Cycles*

Find Features and *Predict Disease*. The following figure shows the phonocardiogram generated by our system and the various buttons for choosing the options.

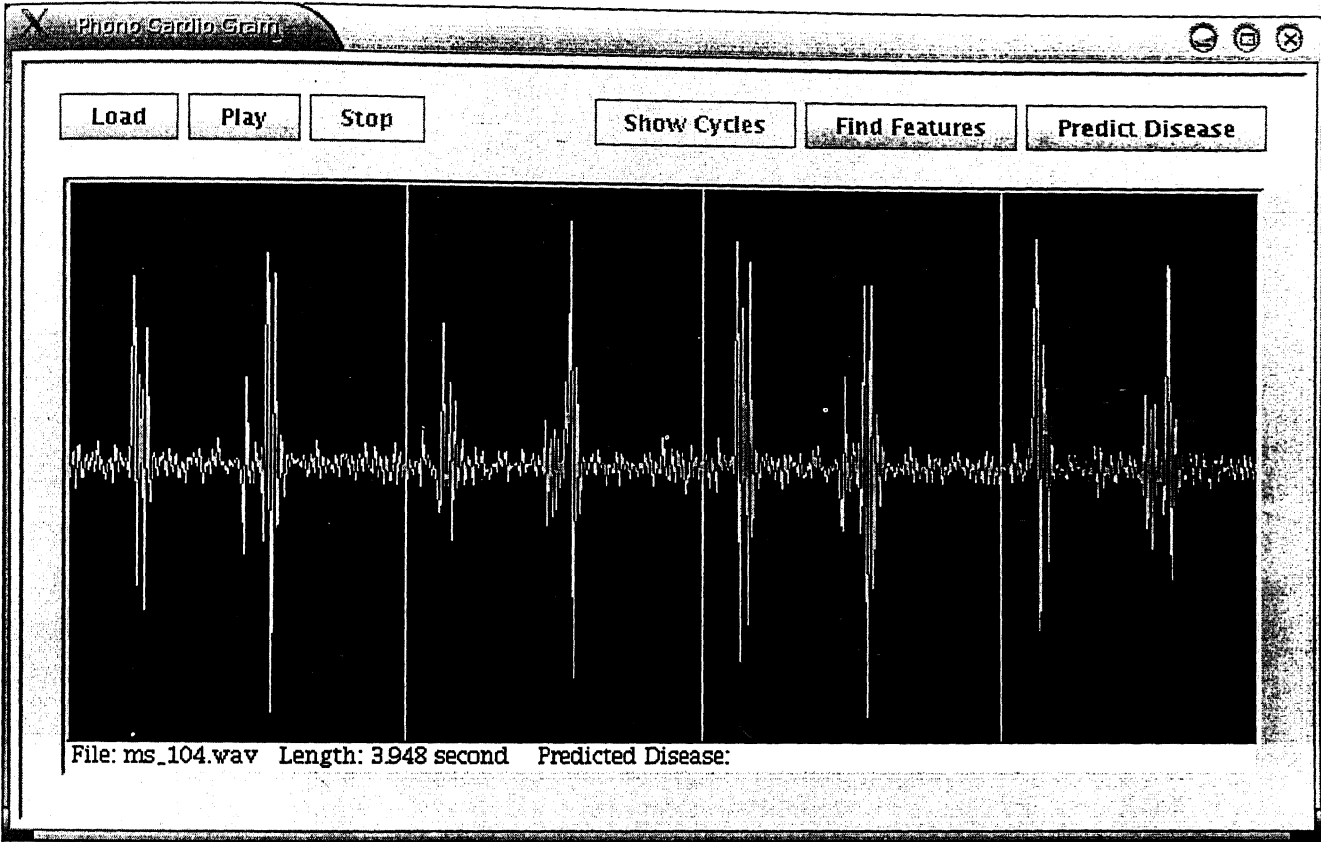


Figure 4.2: The phonocardiogram generated by our system

First, a heart sound file (right now, we are dealing with uncompressed sound file formats, like .wav, .au and .aif) is selected. Then our system reads the data from the file, and plots the data in the phonocardiogram. If the “Play” option is chosen, we can hear the sound of heart beats and a cursor moves over the phonocardiogram, in synchronization with the sound. This is to aid the doctors (or any user), so that they can visualize and hear the sounds simultaneously and repeatedly. The “Show Cycles” option shows the cycles present in the sound sample, with some separating marks. When “Find Features” option is chosen, it extracts some features from a single cycle of heart sound and writes them in a file for further processing. The “Predict Disease” button is used for predicting the category of the particular sound sample. In the next section, we elaborate on how these options work.

4.2 Diagnostic Subsystem

The diagnostic subsystem is implemented in several stages. Here we discuss them one by one.

4.2.1 Data Denoising

The recorded heart sounds often have noise. It is necessary to enhance the signal-to-noise ratio before further processing. Here we adopt a simple data smoothing algorithm called *moving average algorithm*.

Using this, an array of raw (noisy) data $[y_1, y_2, \dots, y_N]$ can be converted to a new array of smoothed data. The "smoothed point" $(y_k)_{smooth}$ is the average of an odd number of consecutive $2n+1$ ($n=1, 2, 3, \dots$) points of the raw data $y_{k-n}, y_{k-n+1}, \dots, y_{k-1}, y_k, y_{k+1}, \dots, y_{k+n-1}, y_{k+n}$. Mathematically,

$$(y_k)_{smooth} = \sum_{i=-n}^{i=n} \frac{y_{k+i}}{2n+1}$$

The odd number $2n+1$ is usually called the filter width. The greater the filter width the more intense is the smoothing effect. This operation is depicted in the figure 4.3.

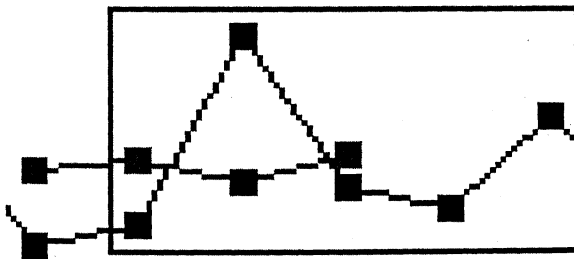
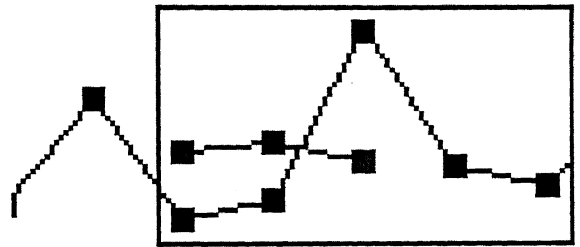
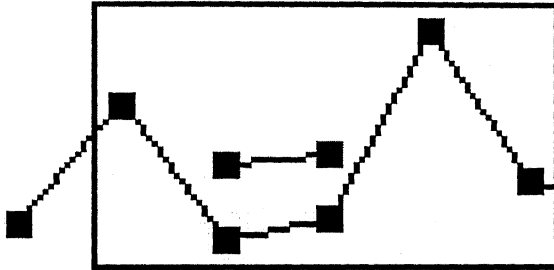
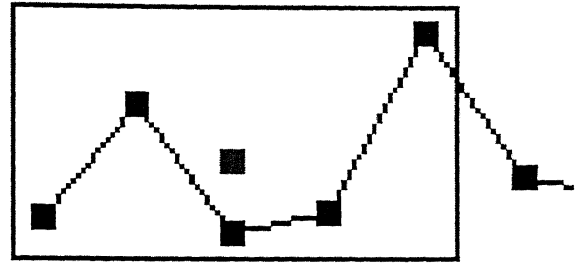
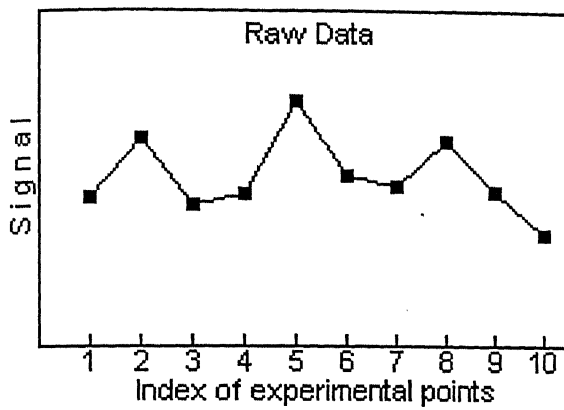
In this example the filter width is 5. The first five raw data (black squares in second figure) within the rectangle (moving window) are averaged and their average value is plotted as smoothed (grey square) data point 3. The rectangle is then moved one point to the right and points 2 through 6 are averaged, and the average is plotted as smoothed data point 4 (third figure). Similar process is carried out for rest of the points. Finally we get a set of smoothed points (last figure). This procedure is called a 5-point unweighted smoothing.

The signal-to-noise ratio may be further enhanced by increasing the filter width or by smoothing the data multiple times. However, with the increase of window width, information may be lost or distorted because too much statistical weight is given to points that are well removed from the central point. To overcome this drawback, we give less weightage to the further points and more weightage to the points near the central point.

4.2.2 Cycle Finding

Our approach extracts each cycle in the sample before the classification step. A normal heart cycle consists of some high activity region, due to the 1st heart sound (S1), followed by a low-activity region. We call this low activity region as *silence period*. Then again there is a high activity region, due to the 2nd heart sound (S2), followed by another silence period. And this is repeated throughout the whole heart sound sample, in a cyclic fashion. Figure 4.4 shows this clearly.

There may be more sound components like third and fourth heart sound (S3 and S4) and corresponding silence periods, but the pattern remains the same from cycle-to-cycle. It is also noticed that the respective durations of these silence periods remains almost same from cycle-to-cycle. We exploit this property to find the cycles.



Smoothed Data

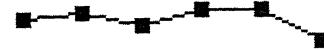


Figure 4.3: Moving average algorithm

Using Silence Period :

Let us call the silence period between S2 and S1 as *alpha* and that between S1 and S2 as *beta*. If there are more heart sounds componets (or murmurs) like S3 and S4, we would get more silence periods like *gamma*, *delta* etc. For the time being let us assume that only S1 and S2 are present, and hence, only *alpha* and *beta* are present. There may be many cycles in a heart sound sample. So, *alphas* and *betas* will also be many in number. Initially, what we do, is to find all the silence periods present in the sample. All those sample points whose absolute data value falls below a threshold, are considered to be part of a silence period. Then, we cluster those silence period lengths using a clustering algorithm (*hierarchichal clustering algorithm*). Ideally, all *alphas* are

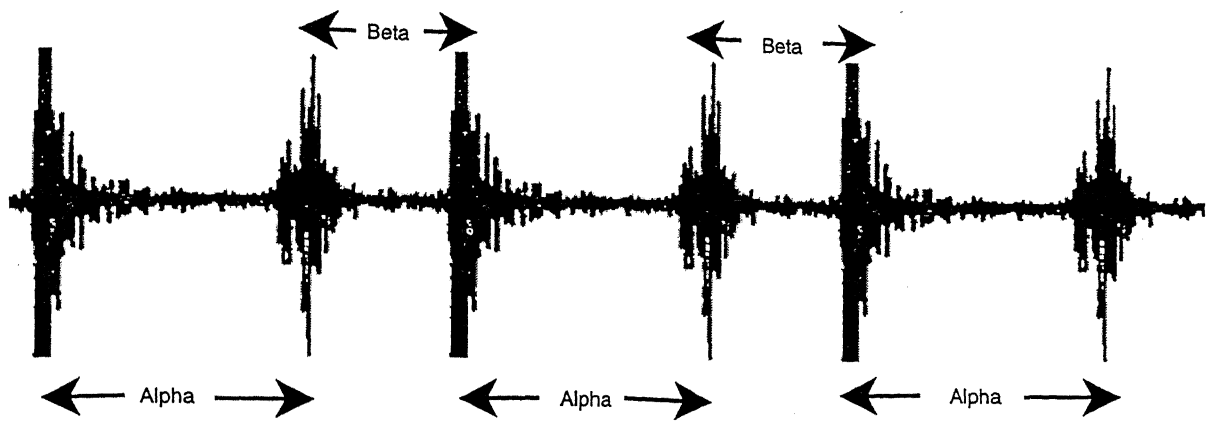


Figure 4.4: The *alpha* and *beta* regions in heart sound

grouped into one cluster and all *betas* into another. Similar is the case if *gamma*, *delta* are present there.

In the next step, we find the longer among *alpha* and *beta*. It has been noticed that, in most of the cases *alphas* are much longer than *betas*. *alphas* are typically $1/2$ of the cycle period, where *betas* are around $1/3$ of the cycle. Next, we parse the whole sample data once again. Whenever a silence period is found, we compare its length with the *alpha* (or *beta*, if *alpha* is not larger) found in the clustering step. If these two are almost equal we label that silence period as a *alpha* region (or *beta*).

In the next step, we compare the distances between the starting of each *alpha* region and the starting of the next *alpha*. If they are almost equal, then we have found the cycles and the cycle durations are the above distances.

Though in most of the cases, this technique works, but it fails in some cases. These situations occur, when noise is very high (of high amplitude) or some unknown spike is found in between a silence period (it divides the silence period, and we get a wrong silence period value) or when *alpha* and *beta* are almost equal in length (then, *alpha* and *beta* would fall in same cluster and we can't differentiate between them). In these cases, we go for the next technique.

Using Signal Cluster :

This step is much like the previous step. The only difference is that, here, instead of clustering the silence period, we cluster the regions with high activity (S1, S2 etc.). The rest of the procedure remains the same. Here again, we check if all the "found cycles" are of almost equal length. If yes, then we have found the cycles. Otherwise, this technique fails too and a cycle can not be found. By experimentation we found that the probability that both techniques fail, is very low.

4.2.3 Feature Extraction

As mentioned in the previous chapter, we use discrete wavelet transform for feature extraction. There is no fixed guideline for choosing the appropriate wavelet. The choice of the appropriate decomposition wavelet depends upon the physiological signal and is chosen empirically. Earlier work suggests that, Daubechies wavelets are best for signals like heart sound, ECG etc and daub4, daub6, daub8 are most frequently used among them. So, we use daub4, daub6 and daub8 wavelet coefficients of the Daubechies family in our work. The same experiment was carried out for these 3 sets of coefficients to see which one gives the best performance.

At first, we sampled 1024 data points from one heart cycle. Then matrix multiplication is done between wavelet coefficient matrix and the data vector (as described in the previous chapter). The matrix is applied using a hierarchical algorithm, called a *pyramidal algorithm*. The wavelet coefficients are arranged so that odd rows contain an ordering of wavelet coefficients that act as the smoothing filter, and the even rows contain an ordering of wavelet coefficients with different signs that act to bring out the details. The matrix is first applied to the original, full-length vector. Then the vector is smoothed and reduced by half and the matrix is applied again. Then the smoothed, halved vector is smoothed, and halved again, and the matrix applied once more. This process continues until one smoothed data and one detailed data remain. Thus, each matrix application brings out a higher resolution of the data while at the same time smoothing the remaining data. The output of the DWT consists of the remaining “smooth” component, and all of the accumulated “detail” components. The above approach yields the required multiresolution analysis. Figure 4.5 explains the above procedure. Here $h(n)$ acts as the smoothing filter and $g(n)$ brings out the detail components.

In the above computation we used the coefficients shown in table 4.1. The coefficients were computed by Daubechies [4, 5].

4.2.4 Feature Selection

The above wavelet decomposition procedure returns as many as 1024 features. This huge number of features will be computationally expensive during the training phase. So, using some feature selection techniques we can discard some features. First, we discard the 4 levels with shortest scale (high frequency, i.e. detailed values). This step leaves us with 64 features and substantially simplifies the neural network for classification. This also reduces noise (as noise is captured in the high frequency decomposition of the wavelet). Next, we take statistics over the set of the wavelet coefficients and use them as features. We take the following statistics :

1. The mean of the absolute value of the coefficients in each subband.

These features provide information about the frequency distribution of the heart sound.

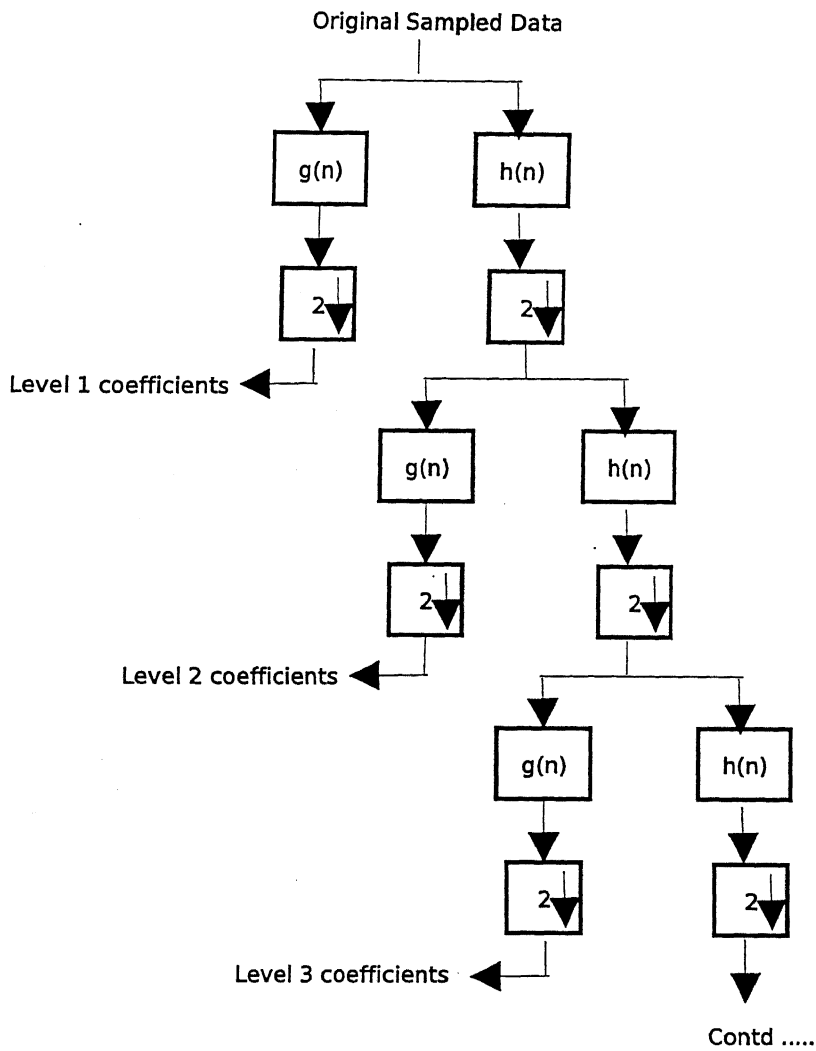


Figure 4.5: Wavelet Decomposition by *pyramidal algorithm*

2. The standard deviation of the coefficients in each subband.

These features provide the information about the amount of change in the frequency distribution.

3. Ratios of the mean values between adjacent subbands.

This feature also provides the information about the frequency distribution.

This results in another 14 features. We use these 78 features (64+14) for classification.

4.2.5 Classification

We use a standard two-layer, fully connected feed-forward network with one hidden layer for classification.

Type	n	Coefficients
D4	0	0.4829629131445341
	1	0.8365163037378079
	2	0.2241438680420134
	3	-0.1294095225512604
D6	0	0.3326705529500825
	1	0.8068915093110924
	2	0.4598775021184914
	3	-0.1350110200102546
	4	-0.0854412738820267
	5	0.0352262918857095
D8	0	0.2303778133088964
	1	0.7148465705529154
	2	0.6308807679398587
	3	-0.0279837694168599
	4	-0.1870348117190931
	5	0.0308413818355607
	6	0.0328830116668852
	7	-0.0105974017850690

Table 4.1: Daubechies' Wavelet Coefficients

Training

The 78 features obtained from each sample are fed to the neural-net for training. So, the number of units in the input layer is 78. We have five categories of heart sounds (*aortic regurgitation, aortic stenosis, mitral regurgitation, mitral stenosis and normal heart sound*). So, the output layer has 5 units. Now, as mentioned earlier, there is no fixed rule for choosing the number of units in the hidden layer. So, we chose the number which gives best results. For our system, we found 78 (the no of input nodes) as the most appropriate size for the hidden layer.

We use the back-propagation algorithm for training the network. However, values of some parameters were chosen by experimentation. These parameters were again chosen empirically, based on the results they gave. Table 4.2 shows the chosen values.

At the end of the training, we get the weight matrix. The matrix is stored in a file, for use during the test process.

The time required for the above training process is around 70 seconds.

Parameter	Value
η	0.01
<i>epoch</i>	2000
n_{input}	78
n_{hidden}	78
n_{output}	5

Table 4.2: Different Parameters

Testing

Testing on an unknown sample is done by extracting the 78 features from the test sample and feeding them as inputs to the trained network. The feed-forward computation is performed, which gives 5 outputs at the 5 output nodes. We choose the node with the highest output value as the prediction. The time required for testing is 200-400 milliseconds.

We also use a calculated value called *confidence* of the prediction. If the difference between the highest output value and the second highest output value is very small (smaller than a threshold), we say that the confidence of the prediction is *low* and we give both the highest and second highest output, as the first and second preference for the predicted disease.

4.3 Results

We collected 56 heart sound samples of five categories (*aortic regurgitation*, *aortic stenosis*, *mitral regurgitation*, *mitral stenosis* and *normal heart sound*). All samples were downloaded from the Internet [23, 24, 25, 26, 27]. To evaluate the performance of our system we used 80-20 method. 80% of the samples, randomly chosen from the 56 samples, were used for training and the rest 20% samples were used for testing. This process was repeated 50 times. The average classification accuracy we obtained for daub4, daub6 and daub8 wavelets are shown in the following table.

Wavelet	Train Samples	Test Samples	Accuracy
<i>daub4</i>	42	14	77 %
<i>daub6</i>			77.14 %
<i>daub8</i>			81.86 %

Table 4.3: Classification Accuracy

We also noticed that, most of those predictions that were wrong, were of "low" confidence.

And the right predictions were present as the second option in these cases.

From the result, it is clear that the performance of daub8 wavelet is better than the other two. So, we use daub8 in the final diagnosis system.

Chapter 5

Conclusion and Future Work

5.1 Conclusion

In our work, we have developed an intelligent heart disease classification system for the prediction of possible heart disease by analysing the heart sound signal. The task of feature extraction was carried out by using wavelet packet decomposition for multi-scale analysis, while the classification was carried out by a feed-forward neural network with back-propagation learning algorithm. We obtained a result of around 81% accuracy in classification. This is much less than the results obtained by other works that we reviewed. Todd R. Reed, Nancy E. Reed and Peter Fritzson [17] achieved 100% accuracy level in classifying heart sounds in 5 categories. Ibrahim Turkoglu, Ahmet Arslan and Ilkay Erdogan [3] obtained almost 94% accuracy in detecting mitral valve diseases. The system developed by Onsy Abdel-Alim, Naddar Hamdy and Mohammed A. E. [16] gave 95% accuracy in the diagnosis process. So, we see that the performance of our system is not very satisfactory. But we must say that the result of this work is promising, considering the fact that sufficient number of training samples were not there to train the system properly. And we automated the cycle extraction process, which failed to detect the actual cycles in some cases. We also faced the following problems, that led to this unsatisfactory performance in many ways.

5.1.1 Problems faced

1. Position of stethoscope during recording gives good information about the heart-sound. But it was not provided in our samples.
2. In some cases, the noise present in the samples was very high.
3. The number of samples were not adequate to train the neural net properly.
4. Information like age, sex of the particular patient were missing. This information is important for a proper diagnosis.

5.1.2 Advantages of our system

The system that we have developed, has following advantages.

1. It is robust enough, to handle noise.
2. The cycles were extracted by the system. So, no need for manually segmenting the cycles present in the sound sample.
3. Our system is rapid, because the time it takes for the testing step, is very low (200-400 milliseconds).
4. It is fully automated, and requires no special skill to handle it.
5. The implementation cost is very less, compared to other techniques (like ECG etc).

5.1.3 Drawbacks of the system

1. Our system may fail if the noise level is very high in the samples.
2. If the cycle-length (in time) and signal-pattern varies a lot from cycle to cycle (which is a very rare case) then our cycle-finding technique would fail.

5.2 Future Work

After getting this moderate accuracy of the system, we should try for achieving higher accuracy, if possible near 100%. For this we should make our cycle-detection technique absolutely fail-proof. Also we need better methods for handling high noise. We also plan to investigate the use of other wavelet families. Another interesting direction is combining features from different analysis techniques.

Another approach, that can be developed, and hopefully would give much better result, is to make a system that deals with a set of techniques like heart sounds from stethoscope, ECG, Doppler ultrasound etc and predict the disease based on the combined result from each of the techniques.

Bibliography

- [1] Tsoi A. C. and Sergejew A. A. An application of artificial neural network technique in ECG signals analysis. *Pan Pacific Workshop on Brain Electric and Magnetic Topography*, Feb 1992.
- [2] Hippenstiel Ralph D. and Monique P. Fargues. Feature extraction from digital communication signals using wavelet transforms. *NPS Technical Report*, Feb 1995.
- [3] Turkoglu Ibrahim, Arslan Ahmet, and Erdogan Ilkay. A pattern recognition system for diagnose of the heart mitral valve diseases based on wavelet packet and neural network. *F.U. Fen ve Muhendislik Dergisi*, 14(1):1-10, 2002.
- [4] Daubechies Ingrid. Orthonormal Bases of Compactly Supported Wavelets. *Communication in pure and applied mathematics*, 41:906-966, 1988.
- [5] Daubechies Ingrid. *Ten Lectures on Wavelets*. SIAM, 1992.
- [6] Ghosh Joydeep, Deuser Larry M., and Beck Steven D. A neural network based hybrid system for detection, characterization and classification of short-duration oceanic signals. *IEEE Journal on Oceanic Engineering*, 17(4), 1992.
- [7] Jacobson M. L. Analysis and classification of physiological signals using wavelet transforms. In *Proceedings of the 10th IEEE International Conference on Electronics, Circuits and Systems*, Dec, 2003.
- [8] Huiying Liang, Sakari Lukkarinen, and Iiro Hartimo. A heart sound segmentation algorithm using wavelet decomposition and reconstruction. In *Proceedings of the Nineteenth International Conference - IEEE/EMBS*, Oct - Nov, 1997.
- [9] Lee Sang Min, Kim In Young, and Hong Seung Hong. Heart sound recognition by new methods using the full cardiac cycled sound data. *IEICE Transaction on Information and System*, E84-D(4), 2001.
- [10] Mitchell Tom M. *Machine Learning*. The McGraw-Hill Companies, Inc.
- [11] Han J. and Kamber M. *Data Mining: Concepts and Techniques*. Morgan Kaufmann Publishers.

- [12] Abdalla S. A. Mohamed and Hazem M. Raafat. Recognition of heart sounds and murmurs for cardiac diagnosis. In *Proceedings of 9th International Conference on Pattern Recognition*, Nov, 1988.
- [13] Abdalla S. A. Mohamed and Hazem M. Raafat. Automatic discrimination between heart sounds and murmurs using parametric models. In *Proceedings of IEEE International Conference on Systems, Man and Cybernetics*, Aug, 1988.
- [14] Hazarika N., Zhu J., Tsoi A. C., and Sergejew. Classification of EEG signals using the wavelet transform. In *Proceedings of 13th International Conference On Digital Signal Processing*, 1997.
- [15] Hazarika N., Zhu J., Tsoi A. C., and Sergejew. Classification of ECG signals using wavelet coefficients and an ANN. *Brain Topography*, 7(99).
- [16] Abdel-Alim Onsy, Hamdy Naddar, and A. E. Mohammed. Heart disease diagnosis using heart sounds. In *Proceedings of the nineteenth national radio science conference, Alexandria*, March, 2002.
- [17] Reed Todd R., Reed Nancy E., and Fritzson Peter. The analysis of heart sounds for symptom detection and machine-aided diagnosis. In *Proceedings of the EUROSIM 2001*, June, 2001.
- [18] Obaidat M. S. and Matalgah M. M. Performance of the short-time Fourier transform and wavelet transform to phonocardiogram signal analysis. In *Proceedings of the ACM/SIGAPP symposium on Applied computing: technological challenges of the 1990's*, 1992.
- [19] Nguyen Chung T., Hammel S. E., and Gong K. F. Wavelet based hybrid neurosystem for feature extraction, characterization and signal classification. *Conference Record of the 29th Asilomar Conference on Signals, Systems and Computers*, 1995.
- [20] <http://www.3m.com>
- [21] <http://www.mamashealth.com>
- [22] <http://www.sci.sdsu.edu>
- [23] <http://www.bioscience.org/atlas/heart>
- [24] <http://www.medstudents.com.br/cardio/heartsounds/heartsou.htm>
- [25] <http://egeneralmedical.com/egeneralmedical/listohearmur.html>
- [26] <http://www.wilkes.med.ucla.edu/inex.htm>
- [27] <http://www.dundee.ac.uk/medther/Cardiology/ms.htm>

[28] <http://www.amara.com/IEEEwave/IEEEwavelet.html>

[29] Wavelet Tutorial. <http://users.rowan.edu/~polikar/WAVELETS/WTtutorial.html>

[30] Texas Heart Institute. <http://www.tmc.edu/~thi>